



Sandia  
National  
Laboratories

Exceptional service in the national interest

## UNLOCKING THE POWER OF LARGE LANGUAGE MODELS: PRACTICAL APPLICATIONS IN SCRIPTS AND PROGRAMS

Anthony Garland

# OUTLINE

- What are LLMs
  - Basic LLM 101
  - Recent advances
- LLMs as part of a system
  - Parts
    - Memory
    - Tools
    - Database
    - LLM
  - Common design patterns
    - Chain of thought /Tree of thought
    - QA over docs
    - Agents

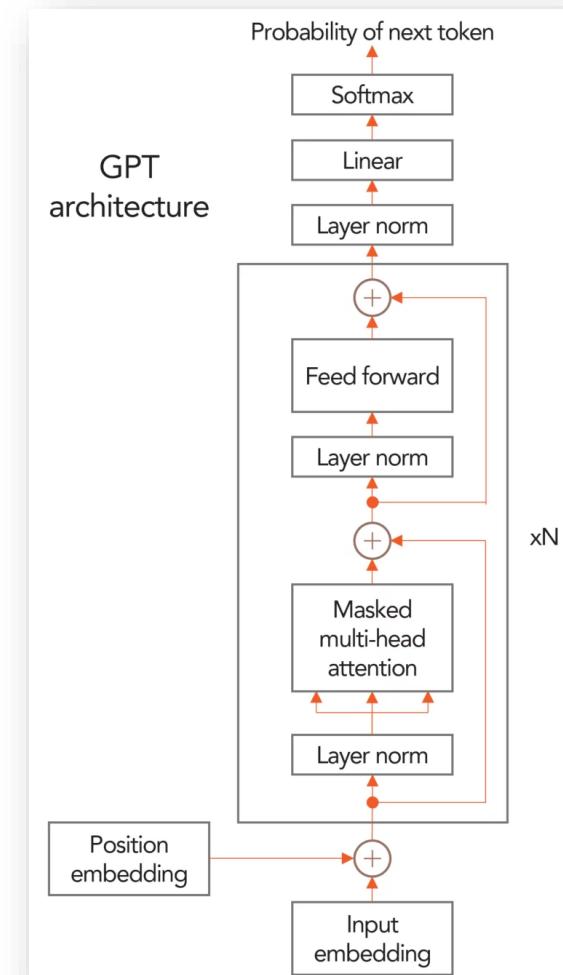
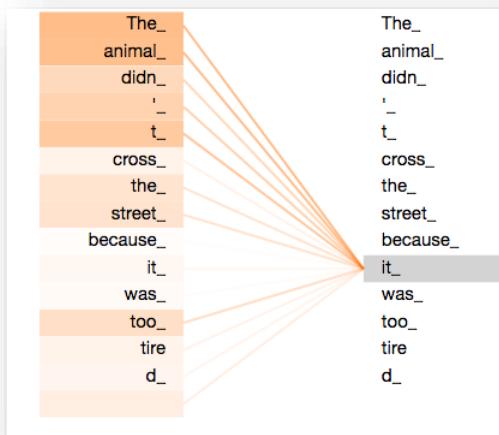




# WHAT ARE LLMS?

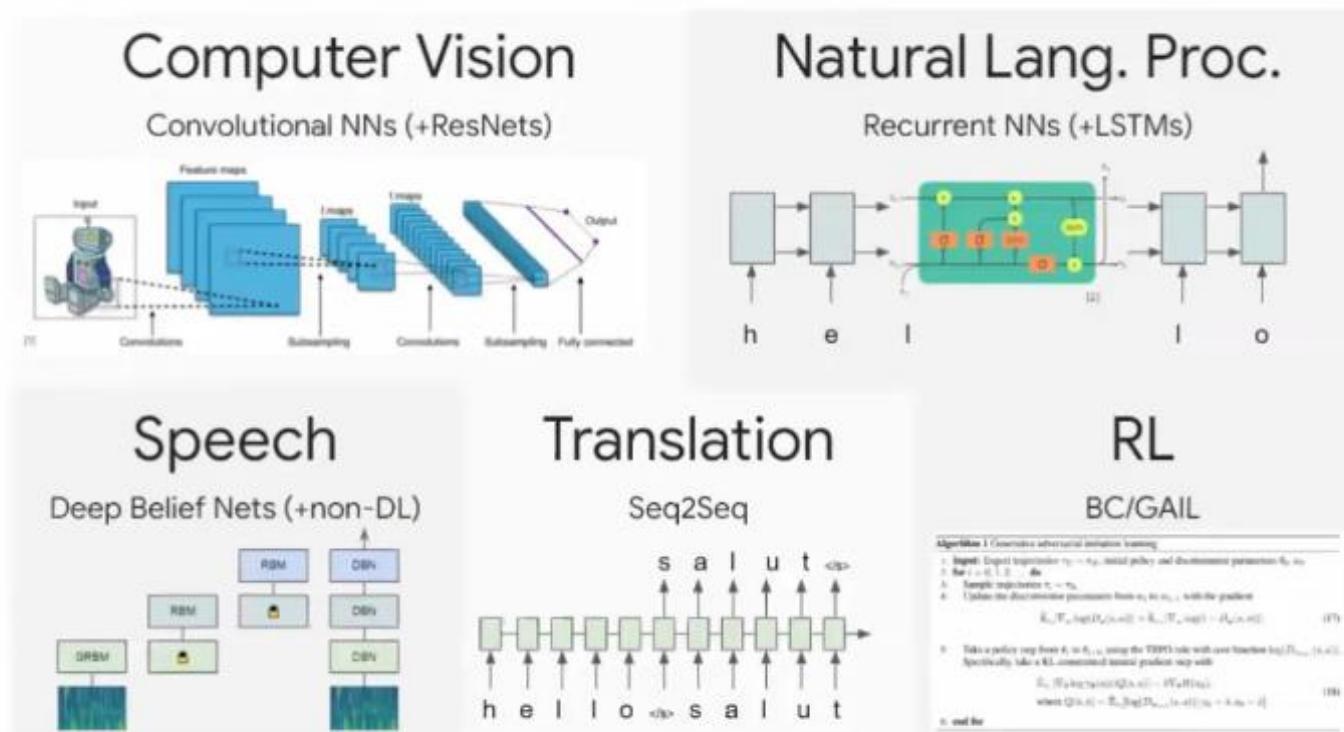
# WHAT ARE LLMS

- Modern LLM are stacked layers of transformers
- Transformers
  - Convert input to a vector (embedding)
  - Use self-attention
    - Scaled dot-product
    - Each word can determine determines how much to pay “attention” to all the other input words
  - Multiple heads. Perform self-attention M times
- Stack transformer layers together N times
- Transformers are a very generic computing paradigm that can use any type of input



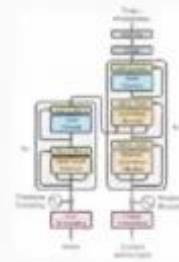
$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

# PRE-TRANSFORMER ML LANDSCAPE

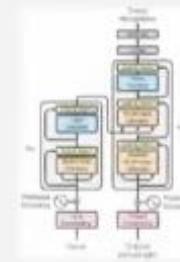


# POST TRANSFORMER ML LANDSCAPE

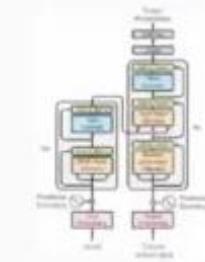
Computer Vision



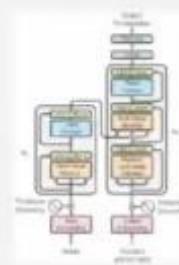
Natural Lang. Proc.



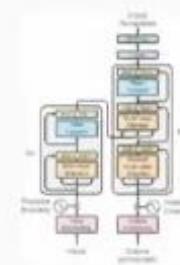
Reinf. Learning



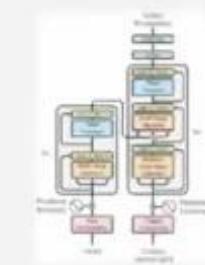
Speech



Translation



Graphs/Science





# WHY INSTRUCT GPT AND CHATGPT WERE INTERESTING

- GPT3 had been around for a while, but it was hard to use.
- Useability!
  - **Alignment** of the output to be helpful to humans
  - The Chat UI worked well.

Example: Write some code to do X

## How to get GPT3 to do what you want

You are GPT-3, and you can't do math.

You can do basic math, and your memorization abilities are impressive, but you can't do any complex calculations that a human could not do in their head. You also have an annoying tendency to just make up highly specific, but wrong, answers.

So we hooked you up to a Python 3 kernel, and now you can execute code. If anyone gives you a hard math problem, just use this format and we will take care of the rest:

Question: \${{Question with hard calculation.}}

```
```python
${{Code that prints what you need to know}}
````
```

Output
\${{Output of your code}}

Answer: \${{Answer}}

Otherwise, use this simpler format:

Question: \${{Question without hard calculation}}

Answer: \${{Answer}}

Begin.

Question: What is 37593 \* 67?

```
```python
print(37593 * 67)
````
```

Output
2518731
````

Answer: 2518731

Question: {question}

## How to get ChatGPT to do what you want



write the python code to calculate 375931\*589



Certainly! Here's a Python code snippet to calculate the product of 375931 and 589:

python

```
result = 375931 * 589
print(result)
```

Copy code

# RECENT ADVANCES IN LLMS

- The “Large” can mean many things
  - Large # parameters (>=7B params [Bits and Bytes paper])
  - Large training data
  - Large in quality of data
- Thriving open source community
- Highly optimized training
- Quantization for running on CPU (or less VRAM on GPU)
  - Use int4 instead of float16 datatypes
- Benchmarks are helpful, but struggling to keep up

**Open LLM Leaderboard**

The Open LLM Leaderboard aims to track, rank and evaluate LLMs and chatbots as they are released. Anyone from the community can submit a model for automated evaluation on the GPU cluster, as long as it is a Transformers model with weights on the Hub. We also support evaluation of models with delta-weights for non-commercial licensed models, such as LLaMa.

Search your model and press ENTER...

LLM Benchmark (lite) Extended view About

Model	Average	ARC (25-s)	HellaSwag (10-s)	MMLU (5-s)	TruthfulQA (MC) (0-s)
<a href="#">tiiuae/falcon-40b-instruct</a>	63.2	61.6	84.4	54.1	52.5
<a href="#">timdettmers/guanaco-65b-merged</a>	62.2	60.2	84.6	52.7	51.3
<a href="#">CalderaAI/39B-Lazarus</a>	60.7	57.6	81.7	45.2	58.3
<a href="#">tiiuae/falcon-40b</a>	66.4	61.9	85.3	52.7	41.7
<a href="#">timdettmers/guanaco-33b-merged</a>	60	58.2	83.5	48.5	50
<a href="#">auboss/llama-30b-supercot</a>	59.8	58.5	82.9	44.3	53.6
<a href="#">huggyllama/llama-65b</a>	58.3	57.8	84.2	48.8	42.3
<a href="#">pinkmanlove/llama-65b-hf</a>	58.3	57.8	84.2	48.8	42.3
<a href="#">llama-65b</a>	58.3	57.8	84.2	48.8	42.3
<a href="#">MetaAI/GPT4-x-Alpaca-30b</a>	57.9	56.7	81.4	43.6	49.7
<a href="#">Aesla/VicunaUnlocked-alpaca-30b</a>	57.6	55	88.8	44	58.4
<a href="#">digitous/Alpaca-30b</a>	57.4	57.1	82.6	46.1	43.8
<a href="#">Aesla/GPT4-x-AlpacaDente2-30b</a>	57.2	56.1	79.8	44	49.1
<a href="#">TheBloke/dromedary-65b-lexa-HF</a>	57	57.8	88.8	59.8	38.8
<a href="#">TheBloke/Wizard-Vicuna-13B-Uncensored-HF</a>	57	53.6	79.6	42.7	52
<a href="#">elinas/llama-30b-hf-transformers-4.29</a>	56.9	57.1	82.6	45.7	42.3
<a href="#">auboss/llama30B-SuperHOT</a>	56.9	57.1	82.6	45.7	42.3
<a href="#">llama-30b</a>	56.9	57.1	82.6	45.7	42.3
<a href="#">cyl/awesome-llama</a>	56.8	54.4	79.7	41.8	51.3
<a href="#">NousResearch/Nous-Hermes-13b</a>	56.4	52.5	88	41.8	51.1
<a href="#">elinas/chronos-33b</a>	56.4	54.9	81.3	43.8	45.8
<a href="#">HuggingFaceHJ/stachat-beta</a>	56.2	52	88.6	44.7	47.3
<a href="#">openaccess-ai-collective/wizard-mega-13b</a>	55.7	52.5	78.6	41	50.6



# LLMS AS PART OF A SYSTEM



# FOUNDATION MODELS

## [2012-2022]

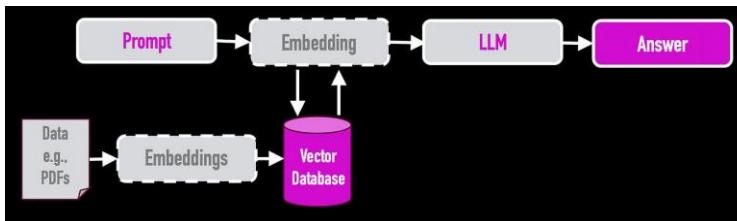
- **Apply DL to task X**
- End-to-End (Given inputs, predict outputs)
- Specific to one task
- Adaptation via fine-tuning only
- Examples
  - BERT
  - CNNs
  - ...

## [2022- now]

- Combine foundation/base models to X,Y,Z to do task Q,R,S
- Can do many tasks
- Adaption via prompting (no parameter updating)
- Powerful building blocks
- Examples
  - LLMs
  - CLIP
  - ...

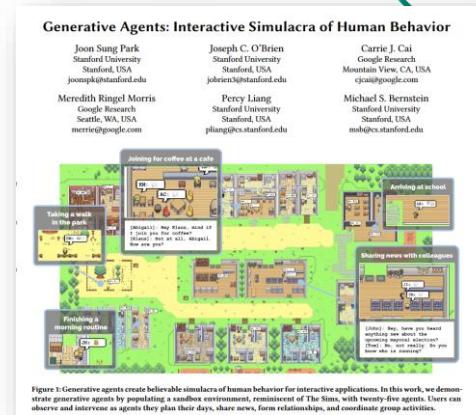
# LLMS AS PART OF A SYSTEM

- ChatGPT and Chat UIs are fun, but basic
- The real power of LLMs comes as part of a system
- What are the building blocks?

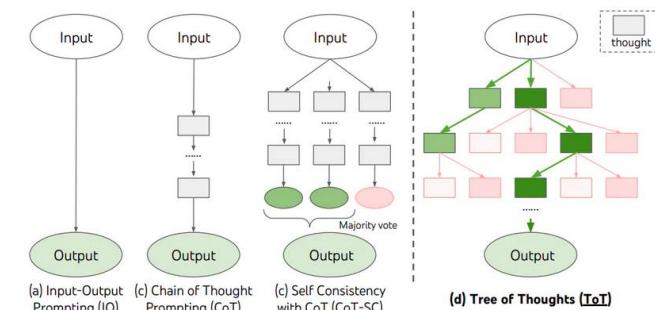
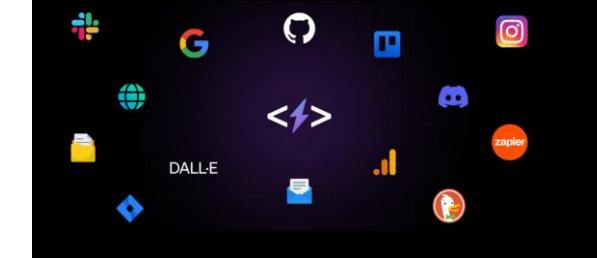


## Building Blocks

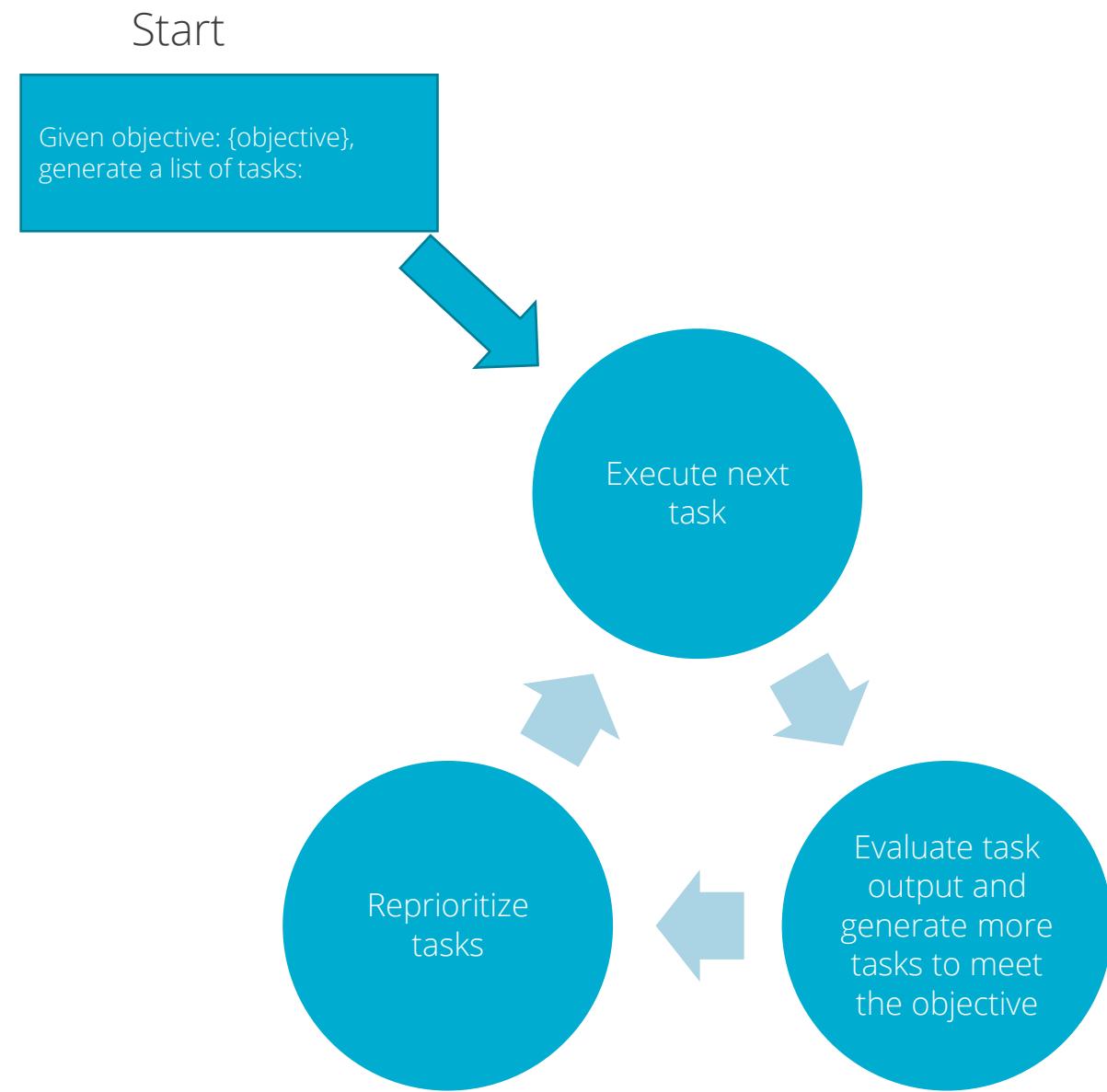
- Memory
  - Long term
  - Short term
- Tools
  - Code interpreters
  - Shells
  - File system
  - API
- Database
- LLM
  - Tree of thought
  - Reflection



### Extend Agent's Capabilities Through Tools



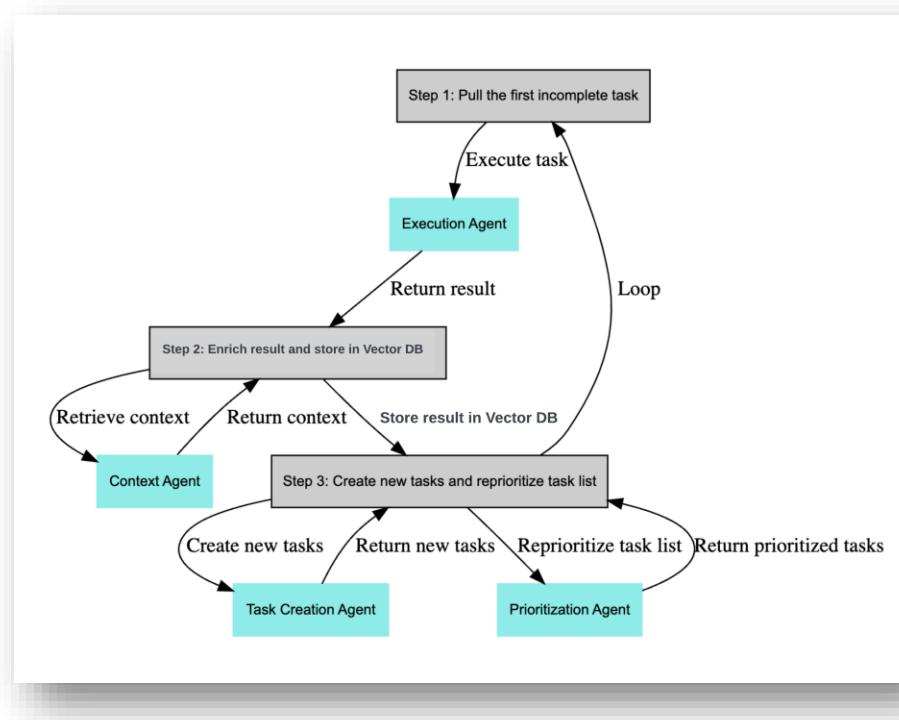
# AGENTS



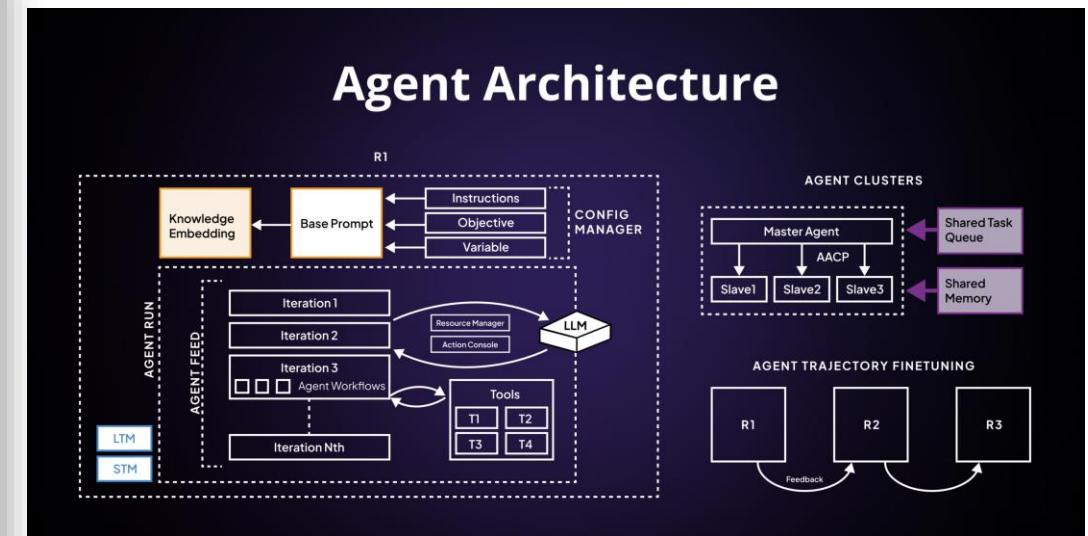
# AGENTS

A few open source projects

1. BabyAGI
2. AutoGPT
3. AgentGPT
4. SuperAGI
5. GPT-engineer
6. ...



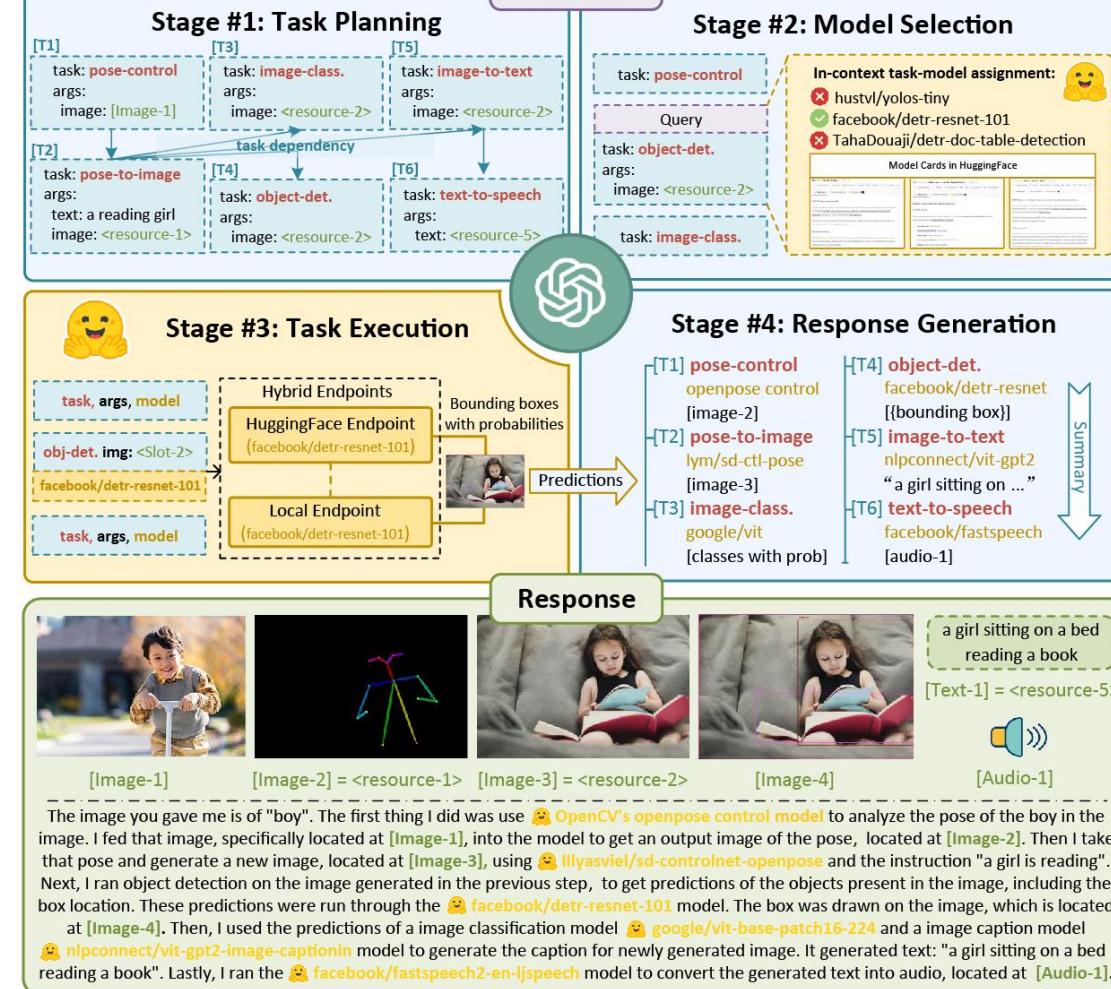
Yoheinakajima BabyAGI



<https://github.com/TransformerOptimus/SuperAGI>

please generate an image where a girl is reading a book, and her pose is the same as the boy in the image example.jpg. Then please describe the new image with your voice.

### Request

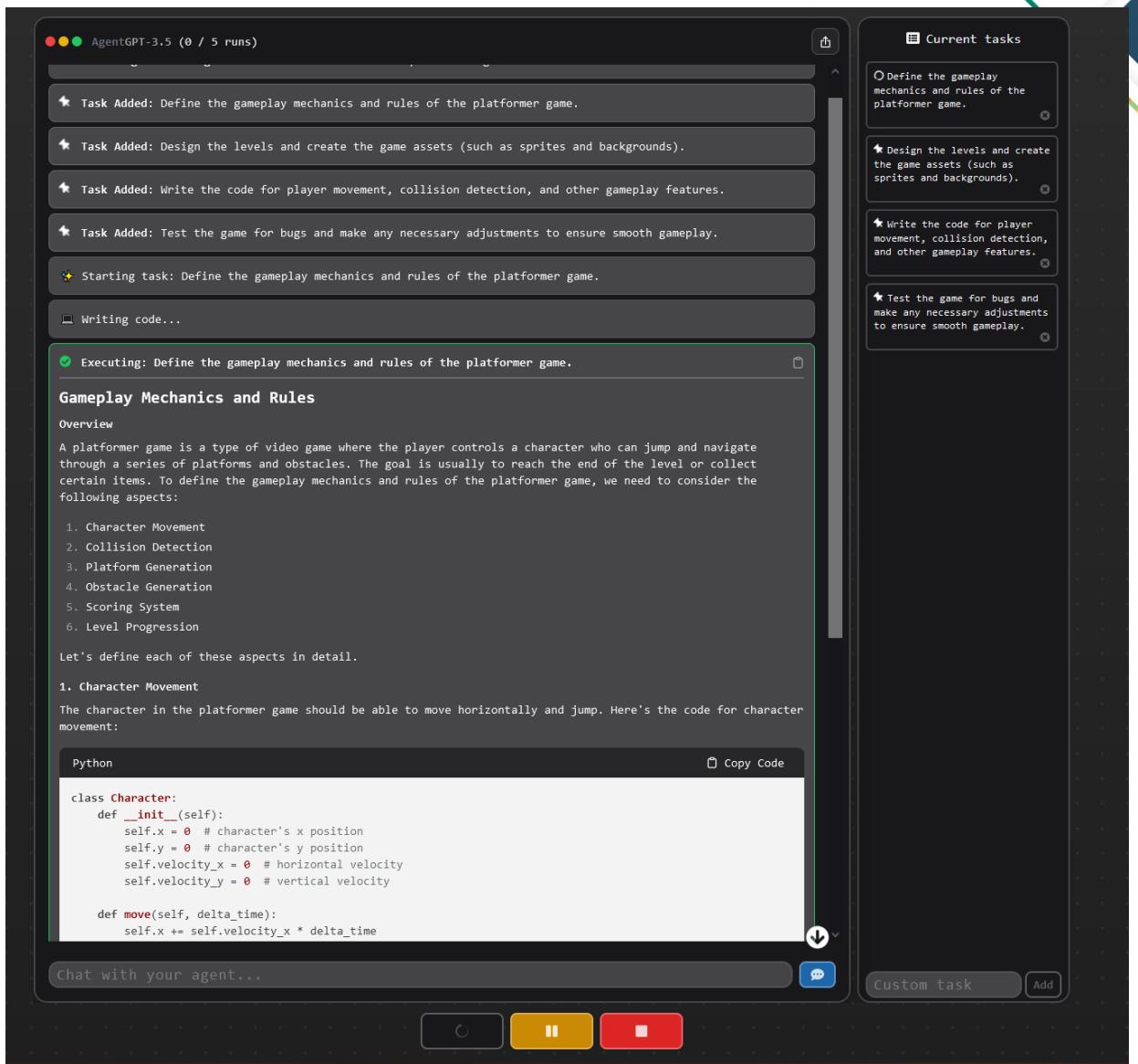


HuggingGPT: Solving AI Tasks with ChatGPT and its Friends in HuggingFace, Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu and Yueting Zhuang

Write some code to make a multiplayer game in python

<https://agentgpt.reworkd.ai/>

<https://github.com/reworkd/AgentGPT>



The screenshot shows the AgentGPT interface. At the top, a task list for "AgentGPT-3.5 (0 / 5 runs)" is displayed, with several "Task Added" items and a "Starting task" item. Below the task list is a "Writing code..." section. A specific task, "Executing: Define the gameplay mechanics and rules of the platformer game," is highlighted. This task leads to a "Gameplay Mechanics and Rules" section with an "Overview" and a list of 6 aspects: 1. Character Movement, 2. Collision Detection, 3. Platform Generation, 4. Obstacle Generation, 5. Scoring System, 6. Level Progression. Below this, a "Let's define each of these aspects in detail." section is shown. Under "1. Character Movement," it says: "The character in the platformer game should be able to move horizontally and jump. Here's the code for character movement." A Python code block is displayed:

```
class Character:
    def __init__(self):
        self.x = 0 # character's x position
        self.y = 0 # character's y position
        self.velocity_x = 0 # horizontal velocity
        self.velocity_y = 0 # vertical velocity

    def move(self, delta_time):
        self.x += self.velocity_x * delta_time
```

At the bottom of the interface, there is a "Chat with your agent..." input field and a set of control buttons (play, pause, stop).

# AgentGPT templates available now



<b>ResearchGPT</b> Academics and Professional Generate a thorough report on a specific subject	<b>BrandGPT</b> Academics and Professional Evaluate a brand's performance, market position, and future prospects	<b>TravelGPT</b> Other Plan a detailed journey to a selected destination	<b>PlatformerGPT</b> Creative and Social Code a platformer game featuring a popular character or theme
<b>IndustryGPT</b> Academics and Professional Present a comprehensive review of an industry, covering key trends, players, and future predictions	<b>ScraperGPT</b> Other Extract and summarize data from a selected website	<b># PostGPT</b> Creative and Social Create engaging captions and hashtags for your social media posts	<b>EmailGPT</b> Academics and Professional Compose a concise and detailed email
<b>ResumeGPT</b> Academics and Professional Design a professional resume based on your career history and skills	<b>NovelGPT</b> Creative and Social Begin writing a novel in a selected genre	<b>DietGPT</b> Health and Fitness Create a customized diet plan based on dietary preferences and goals	<b>FitnessGPT</b> Health and Fitness Design a workout regimen tailored to your fitness goals
<b>MarketingGPT</b> Academics and Professional Design a comprehensive marketing strategy for your business	<b>BudgetGPT</b> Academics and Professional Prepare a personal or family budget	<b>StudyGPT</b> Academics and Professional Design a study schedule to achieve your academic objectives	<b>NewsGPT</b> Other Author a detailed news article on a selected topic
<b>EventPlannerGPT</b> Other Organize a detailed schedule for your forthcoming event	<b>BlogGPT</b> Creative and Social Write a blog post on a selected topic	<b>AstroGPT</b> Science and Technology Discuss astronomical phenomena, discoveries, and related technology	<b>ArtReviewGPT</b> Creative and Social Critique a piece of art, discussing its style, context, and influence

SNL can make their own

**EngineerGPT**  
Assist an engineer in designing a 3D object

**ScienceGPT**  
Analyze this data and write a report

**SierraGPT**  
Design a sierra input deck based on these requirements and 3d models

**Sierra2GPT**  
Interpret the output of sierra to see if it meets these requirements

**ContractGPT**  
Design a WPA with another agent to get job X done.

**SummaryGPT**  
Given these historical documents and these new concerns, write a report

## CONCLUSION

- Chatbots are cool, but aren't why LLMs are so powerful
- LLMs as a system are powerful
  - LLM
  - Controller
  - Synthesizer
- Tools provide the system with more capabilities
  - Good systems are Self-correcting
- The exact setup of how to piece together different building blocks is an open question
- All the tools exist and open source right now!
  - Open source LLMs
  - Open source tool kits
- How to get started?
  - LangChain
  - LLamaIndex
  - AgentGPT
- Agents will only get better with time and research



**LangChain**

